**Explanation\_Q6: CRUD Operations using ArrayList**

### Objective:

To implement CRUD (Create, Read, Update, Delete) operations using the ArrayList collection in Java, specifically on a custom Employee class.

### 1. Employee Class Overview:

class Employee {

private int id;

private String name;

private String department;

// Constructor, Getters, Setters, and toString() methods

}

**Encapsulates** employee data with fields: id, name, department

### 2. CRUD Implementation Using ArrayList:

ArrayList<Employee> employeeList = new ArrayList<>();

This list stores all employee objects in memory and is manipulated as follows:

#### A. Create (Add Employee)

Instantiate a new Employee object

Add it to the employeeList using add()

#### B. Read (Display All Employees)

Use a loop (e.g., for-each) to iterate and print each employee's details

#### C. Update (Modify Employee Details)

Search for an employee by id

Update the name or department fields using setters

#### D. Delete (Remove an Employee)

Search by id and remove the employee using remove()

### 3. Example Output :

Suppose you:

Add 3 employees

Update 1 employee's department

Delete 1 employee by ID

**Expected flow:**

Employee added: [ID: 101, Name: Alice, Dept: HR]

Employee added: [ID: 102, Name: Bob, Dept: IT]

Employee added: [ID: 103, Name: Charlie, Dept: Finance]

Before Update:

... (All employees listed)

Updated Employee ID 102's department to Marketing

After Update:

... (Updated list)

Deleted Employee with ID 103

Final List:

... (Remaining 2 employees)

